

REMARKS

The Examiner is thanked for his careful and thorough Office Action.

Claims 1-18 are pending in the present application. All claims were rejected.

Reconsideration of the claims is respectfully requested.

Claim Rejections -- 35 U.S.C. § 102

Claims 1-5 and 8-16 were rejected under 35 U.S.C. ' 102(e) as being anticipated by *Perkowski* (US Patent Publication 2003/0139975, hereinafter *Perkowski*). This rejection is respectfully traversed.

As many of the claims depend, directly or indirectly, from claim 1, identifying a distinction in claim 1 over *Perkowski* will demonstrate that that *Perkowski* does not anticipate claim 1 or any of its dependents. To support the anticipation rejection, it is important that *Perkowski* be shown not only to include every claimed element, but that those elements interact as claimed. Some of the many claim distinctions are discussed below.

Claim 1 requires, in part, “a first computer subsystem comprising collaborative application software ... for sending application output data over the computer network ... and ... a second-subsystem firewall, located in front of the second application subsystem, ... to communicate the application output data to the second computer subsystem.”

These limitations are not taught or suggested by *Perkowski*. For example, the only real description of *Perkowski*'s “Collaborative Replenishment System” is found in paragraph 185, in relation to figures 2-1 and 2-2. Nothing in this description, these figures, or anywhere else in

Perkowski teaches or suggests that the Collaborative Replenishment System sends output data over a network, through a second-subsystem firewall, to a second computer subsystem.

While Perkowski does mention a firewall in several instances, none of these are with regard to the Collaborative Replenishment System. A firewall is discussed with regard to figures 2C and 2D, as part of the client systems of figures 2-1 and 2-2. Paragraph 185 appears to state that the client systems 13 of these figures are themselves a part of the Collaborative Replenishment system, so they cannot a part of the claimed second computer subsystem. For this reason alone, claim 1, and all its direct and indirect dependents, should be allowed over Perkowski.

Further, claims 1 requires “the second-subsystem firewall structured to communicate the application output data to the second computer subsystem through a hypertext transfer protocol keep-alive connection that is kept open for the duration of a collaboration.” Claims 11, 15, and the claims that depend from claim 15 similarly require a keep-alive connection. Nothing in Perkowski teaches or suggests an HTTP keep-alive connection at all. As the Examiner is surely aware (and as discussed in part in Erickson, below), a keep-alive connection has a specific meaning, referring to a specific type of HTTP connection, and has only been supported in later versions of the HTTP specifications. Nothing in Perkowski teaches or suggests this type of connection (which is not believed to have even existed at the time of filing of earlier applications in Perkowski’s chain-of-priority), and the term “alive” is does not even appear in Perkowski. The extant Office Action references paragraph 178 for this limitation, apparently for the “dedicated Internet connection” language, but a dedicated physical Internet connection is not at all the same as an HTTP keep-alive connection.

In light of this additional distinction, claims 1 and 15, and all of their direct and indirect dependents, should be allowed over Perkowski.

Nothing in Perkowski teaches or suggests “opening a first-subsystem thread in the second computer subsystem” as in Claim 2. The only reference to a thread at all appears in paragraph 764, and this does not meet the claim limitation. The undersigned as studied paragraph 163 as referenced by the Office Action, but it appears irrelevant to the claim language. Claim 2, and all other claims referencing “threads,” should be allowed over Perkowski.

With regard to claims 3, 10, 12, and 16, nothing in Perkowski teaches or suggests a socket that blocks on a read. “Block on a read” generally means that when a “read” process is used on a socket, any other threads or processes are “blocked” from accessing the socket. This has nothing to do with “performing a search,” and the undersigned is mystified at why the Examiner indicates that “block on a read” reads on “performing a search.” The Examiner is respectfully requested to provide documentary support for this interpretation. All of these claims should be allowed over Perkowski.

With regard to claim 4, the Office Action appears to equate “causing the first-subsystem thread to sleep” with an entire client system being in an “idle” mode because it is not actively being used. These are entirely different concepts, and as nothing in Perkowski teaches or suggests the operation of threads at all, certainly nothing teaches or suggests putting a thread to sleep. Claim 4 should be allowed over Perkowski.

With regard to claim 9, Perkowski’s Collaborative Replenishment System does not appear to send data to a Web server over a second computer network and from there to a second computer subsystem over a first computer network. If the applicant has misread the operation of the

Collaborative Replenishment System, the Examiner is respectfully requested to show where in Perkowski these limitations are met, as it does not appear to be so described in the paragraphs cited in the Office Action.

With regard to claim 14, Perkowski does not teach or suggest that its Collaborative Replenishment System is a word processor, a task scheduling tool, a graphics program, a spreadsheet, a game, or a music studio. The only function described for the Collaborative Replenishment System is “for determining what products retailers can buy in order to satisfy consumer demand at any given point of time.” Claim 14 should be allowed over Perkowski.

Applicant further notes that “collaborative” is described in the instant application as “wherein two or more mutually-remote clients concurrently and simultaneously access and control an application (e.g., a word processing application on a remote server machine) over a computer network across one or more firewalls.” A “collaborative application” is defined in the specification as “an application capable of concurrently receiving input from and providing output to at least two people at two different computers.” Although Perkowski uses the term “collaborative,” nothing in Perkowski or Erickson, below, appear to teach or suggest the use of collaborative application software as described, defined, and claimed in the present application.

Claim Rejections -- 35 U.S.C. § 103

Claims 6-7 and 17-18 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Perkowski in view of *Erickson et al.* (USP 6,412,009, hereinafter Erickson). This rejection is respectfully traversed.

As noted above, nothing in Perkowski or Erickson, or any combination of them, appears to teach or suggest the use of collaborative application software as described, defined, and claimed in independent claims 1 and 15 of the present application. As such, these independent claims, and all dependent claims (including claims 6-7 and 17-18) should be allowed over Perkowski and Erickson.

All anticipation and obviousness rejections have been traversed, and reconsideration and allowance of all claims is respectfully requested.

SUMMARY

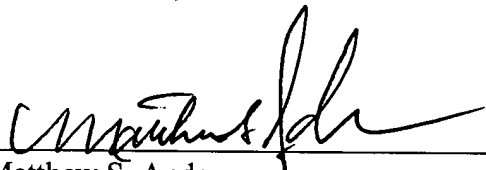
If any issues arise, or if the Examiner has any suggestions for expediting allowance of this Application, the Applicant respectfully invites the Examiner to contact the undersigned at the telephone number indicated below or at *manderson@davismunck.com*.

The Commissioner is hereby authorized to charge any additional fees connected with this communication or credit any overpayment to Deposit Account No. 05-0765.

Respectfully submitted,

DAVIS MUNCK, P.C.

Date: 12/11/03


Matthew S. Anderson
Registration No. 39,093

P.O. Drawer 800889
Dallas, Texas 75380
(972) 628-3600 (main number)
(972) 628-3616 (fax)
E-mail: *manderson@davismunck.com*